



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

10/809,152

03/24/2004

Patrick L. Edson

MWS-104RCE3

7394

74321

7590

04/13/2011

NELSON MULLINS RILEY & SCARBOROUGH/THE MATHWORKS
FLOOR 30, SUITE 3000
One Post Office Square
Boston, MA 02109-2127

EXAMINER

ALVESTEFFER, STEPHEN D

ART UNIT

PAPER NUMBER

2171

MAIL DATE

DELIVERY MODE

04/13/2011

PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary

Application No.

10/809,152

Applicant(s)

EDSON ET AL.

Examiner

Stephen Alvesteffer

Art Unit

2171

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 16 August 2010.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1,3-10,12-36,38-50 and 52-59 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1,3-10,12-36,38-50 and 52-59 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
- ☐ Certified copies of the priority documents have been received.
 - ☐ Certified copies of the priority documents have been received in Application No. _____.
 - ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

Response to Amendment

This Office Action is responsive to the Request for Continued Examination (RCE) filed August 16, 2010. Claims 1, 30, 44, and 59 are amended. Claims 2, 11, 37, and 51 were previously cancelled. Claims 1, 30, 44, and 59 are independent. Claims 1, 3-10, 12-36, 38-50, and 52-59 remain pending.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

Claims 1, 3-6, 25, 28, 56, and 59 are rejected under 35 U.S.C. 103(a) as being unpatentable over Carlson et al. (hereinafter Carlson), United States Patent Application Publication 2003/0033398, and Johnson et al. (hereinafter Johnson), United States Patent Application Publication 2003/0001896.

Regarding claim 1, Carlson substantially teaches a computer readable storage medium storing computer executable instructions that when executed on a processor manage a graphical interface, the medium storing:

instructions for providing a graphical interface (see Carlson paragraph [0087], "FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the

resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11"), where the graphical interface:

accesses a hardware device that is associated with a plurality of properties used to communicate with the hardware device (see Carlson paragraph [0022], "*Further provided is a method, system, and program for configuring multiple resources in the system. User selection is received of one of multiple configuration policies, wherein each configuration policy defines resources to configure and one element for each resource to configure, and wherein each element specifies configuration parameters to use to configure the resource.*"), and

accesses a software device being accessible through the graphical interface, the software device being accessible to a computer, the hardware device associated with a plurality of properties used to communicate with the hardware device (see Carlson paragraph [0039], "*The services may include hardware devices, software devices, application programs, storage resources, communication channels, etc.*");

instructions for providing a first interactive hardware object, where the first interactive hardware object: is accessible to the computer, is depicted in the graphical interface, and interacts with the hardware device (see Carlson paragraph [0087], "*FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11. GUI panel 650 displays icons representing each resource selected from the*

Art Unit: 2175

previous panel 600 and an associated drop down menu 652, 654, 656, 658 for the selected resources.”);

instructions for providing a first configuration represented by the first interactive hardware object, where the first configuration represents a collection of properties used to communicate with the hardware device and a first collection of values associated with the properties (see Carlson paragraph [0087], *“For instance, FIG. 12 illustrates the storage device drop down menu 658 showing four different possible predefined storage device element configurations, including the selected configuration. The other droop down menus 654 and 656 show user or default selected element configurations to include in the service configuration policy being defined. The user would use the drop down menus 652, 654, 656, and 658 to select one predefined element configuration for each selected resource to add to the service configuration policy.”);*

instructions for providing a second interactive hardware object, where the second interactive hardware object: is accessible to the computer, is depicted in the graphical interface, and interacts with the hardware device (see Carlson paragraph [0087], *“FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11. GUI panel 650 displays icons representing each resource selected from the previous panel 600 and an associated drop down menu 652, 654, 656, 658 for the selected resources.”);*

instructions for providing a second configuration represented by the second interactive hardware object, where the second configuration represents the same collection of properties as the first configuration and a second collection of values associated with the properties, wherein at least one value of a property differs between the first configuration and the second configuration (see Carlson paragraph [0087], *“For instance, FIG. 12 illustrates the storage device drop down menu 658 showing four different possible predefined storage device element configurations, including the selected configuration. The other drop down menus 654 and 656 show user or default selected element configurations to include in the service configuration policy being defined. The user would use the drop down menus 652, 654, 656, and 658 to select one predefined element configuration for each selected resource to add to the service configuration policy.”*);

instructions for providing a software object, wherein the software object is representative of the software device, where the software object is depicted in the graphical interface and is configured to be interactive with the software device (Johnson, addressed below);

instructions for displaying the first hardware object and the second hardware object simultaneously (see Carlson paragraph [0087], *“FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11. GUI panel 650 displays icons*

Art Unit: 2175

representing each resource selected from the previous panel 600 and an associated drop down menu 652, 654, 656, 658 for the selected resources.”);

instructions for receiving, from a user, a selection of one hardware object (see Carlson paragraph [0087], “*FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11. GUI panel 650 displays icons representing each resource selected from the previous panel 600 and an associated drop down menu 652, 654, 656, 658 for the selected resources.*”); and

instructions for communicating with the hardware device corresponding to the selected hardware object using the configuration represented by the hardware object (see Carlson paragraph [0087], “*Each drop down menu 652, 654, 656, 658 presents available element configurations 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c the user may select to add to the service configuration policy. As discussed, the element configurations enable configuration and control over a resource according to a predefined level.*”).

Carlson does not explicitly disclose instructions for providing a software object, wherein the software object is representative of the software device, where the software object is depicted in the graphical interface and is configured to be interactive with the software device. Johnson teaches a measurement system graphical user interface for configuring measurement applications including software devices (see Johnson paragraph [0107], “*The run-time builder may also provide various parameters to*

*hardware and/or software resources or devices comprised in the system to configure the hardware and/or software devices in the system according to the run-time specification to allow these devices to be used during execution of the run-time 790"). It would have been obvious to one having ordinary skill in the art at the time the invention was made to provide graphical software device configuration as taught by Carlson in the configuration invention of Carlson so that software devices may also be configured in addition to the hardware devices. Carlson acknowledges that users may need to access software devices in addition to the hardware devices (see Carlson paragraph [0039], "*The services may include hardware devices, software devices, application programs, storage resources, communication channels, etc*").*

Regarding claim 3, Carlson/Johnson teaches providing an analysis object, wherein said analysis object is adapted to communicate with at least one of said hardware object and said software object for analysis of data from at least one of said hardware object and said software object (see Johnson paragraph [0255]; "*Upon execution of the graphical program, the node may receive the measurement task specification as input, invoke an expert system to analyze the measurement task specification and generate a run-time specification for the measurement task in response to the analyzing, as shown in 750 and 770 of FIG. 12*").

Regarding claim 4, Carlson/Johnson teaches instructions for receiving code for execution by the hardware object (see Johnson paragraph [0099]; "*underlying program instructions and/or data structures which are executed by a processor (or programmable hardware element...*").

Regarding claim 5, Carlson/Johnson teaches that a plurality of hardware objects are provided for a single hardware device (see Carlson paragraph [0087]; “*FIG. 12 illustrates the storage device drop down menu 658 showing four different possible predefined storage device element configurations, including the selected configuration*”).

Regarding claim 6, Carlson/Johnson teaches that a plurality of hardware objects are provided for a plurality of hardware devices (see Carlson paragraph [0087]; “*FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11. GUI panel 650 displays icons representing each resource selected from the previous panel 600 and an associated drop down menu 652, 654, 656, 658 for the selected resources*”).

Regarding claim 25, Carlson/Johnson teaches that the graphical interface is implemented with an extensible API (see Carlson paragraph [0044], “*The resource application program interfaces (APIs) 126 provide access to the configuration functions of the resource to perform the resource specific configuration operations*”).

Regarding claim 28, Carlson/Johnson teaches that the graphical interface is adapted to operate on a plurality of operating systems (see Carlson paragraph [0040], “*In Jiro, the service proxy object is written in the Java** programming language, and includes methods and interfaces to allow users to invoke and execute the service object located through the lookup service*”, Java applications are well known in the art as being cross-platform).

Regarding claim 56, Carlson/Johnson teaches that the hardware object enables communication between the graphical interface and the hardware device, and the software object enables communication between the graphical interface and the software device (see Johnson paragraph [0125]; *“a graphical user interface (GUI) may be displayed which presents information for guiding the user in specifying a measurement task. The measurement task may involve a simple measurement using a single instrument or device, or may comprise a complex measurement operation using a plurality of measurement devices. In one embodiment, at least one of the plurality of measurement devices may comprise a measurement hardware device. In another embodiment, at least one of the plurality of measurement devices may comprise a virtual measurement device”*).

Regarding claim 59, Carlson/Johnson teaches a computer readable storage medium storing computer executable instructions that when executed on a processor manage a graphical interface, the medium storing:

instructions for providing a graphical interface (see Carlson paragraph [0087], *“FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11”*), at least one hardware device and one software device being accessible through the graphical interface, the graphical interface being updated in response to a change in the hardware device or the software device, the hardware device associated with a plurality of properties used to communicate with the hardware device (see Carlson

Art Unit: 2175

paragraph [0022], *“Further provided is a method, system, and program for configuring multiple resources in the system. User selection is received of one of multiple configuration policies, wherein each configuration policy defines resources to configure and one element for each resource to configure, and wherein each element specifies configuration parameters to use to configure the resource.”*; see also Carlson paragraph [0039], *“The services may include hardware devices, software devices, application programs, storage resources, communication channels, etc.”*);

instructions for providing a first hardware object, where the first hardware object: is accessible to the computer, is depicted in the graphical interface, and interacts with the hardware device (see Carlson paragraph [0087], *“FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11. GUI panel 650 displays icons representing each resource selected from the previous panel 600 and an associated drop down menu 652, 654, 656, 658 for the selected resources.”*);

instructions for providing a first configuration of the hardware device represented by the first hardware object, the first configuration representing a collection of properties used to communicate with the hardware device and a first collection of values associated with the properties (see Carlson paragraph [0087], *“For instance, FIG. 12 illustrates the storage device drop down menu 658 showing four different possible predefined storage device element configurations, including the selected configuration. The other droop down menus 654 and 656 show user or default selected element*

Art Unit: 2175

configurations to include in the service configuration policy being defined. The user would use the drop down menus 652, 654, 656, and 658 to select one predefined element configuration for each selected resource to add to the service configuration policy.”);

instructions for providing a second hardware object, where the second hardware object: is accessible to the computer, is depicted in the graphical interface, and interacts with the hardware device (see Carlson paragraph [0087], “FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11. GUI panel 650 displays icons representing each resource selected from the previous panel 600 and an associated drop down menu 652, 654, 656, 658 for the selected resources.”);

instructions for providing a second configuration of the hardware device represented by the second hardware object, the second configuration representing the same collection of properties as the first configuration and a second collection of values associated with the properties, wherein at least one value of a property differs between the first configuration and the second configuration (see Carlson paragraph [0087], “For instance, FIG. 12 illustrates the storage device drop down menu 658 showing four different possible predefined storage device element configurations, including the selected configuration. The other drop down menus 654 and 656 show user or default selected element configurations to include in the service configuration policy being defined. The user would use the drop down menus 652, 654, 656, and 658 to select one

Art Unit: 2175

predefined element configuration for each selected resource to add to the service configuration policy.”);

instructions for providing a plurality of software objects, each representative of a software device accessible to the computer, where each of the software objects is depicted in the graphical interface and is configured to be interactive with the software device (see Johnson paragraph [0107], “*The run-time builder may also provide various parameters to hardware and/or software resources or devices comprised in the system to configure the hardware and/or software devices in the system according to the run-time specification to allow these devices to be used during execution of the run-time 790*”);

instructions for displaying the first hardware object, the second hardware object, and the plurality of software objects to a user in a single graphical interface simultaneously (see Carlson paragraph [0087], “*FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11. GUI panel 650 displays icons representing each resource selected from the previous panel 600 and an associated drop down menu 652, 654, 656, 658 for the selected resources.*”);

instructions for receiving, from a user, a selection of at most one hardware object (see Carlson paragraph [0087], “*FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the*

previous GUI panel 600 in FIG. 11. GUI panel 650 displays icons representing each resource selected from the previous panel 600 and an associated drop down menu 652, 654, 656, 658 for the selected resources.”); and

instructions for communicating with the hardware device corresponding to the selected hardware object using the first configuration or the second configuration (see Carlson paragraph [0087], “*Each drop down menu 652, 654, 656, 658 presents available element configurations 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c the user may select to add to the service configuration policy. As discussed, the element configurations enable configuration and control over a resource according to a predefined level.*”).

Claims 7, 8, and 12-14 are rejected under 35 U.S.C. 103(a) as being unpatentable over Carlson (US 2003/0033398) *supra*, Johnson (2003/0001896) *supra*, and Fuller, III et al. (hereinafter Fuller), United States Patent Application Publication number 2003/0035008.

Regarding claim 7, Carlson/Johnson teaches every limitation of claim 7 except instructions for scanning for available hardware; and instructions for creating an additional hardware object for each hardware device detected and not already associated with a hardware object. Fuller teaches a method and apparatus for controlling an instrumentation system that automatically scans for available hardware (instruments) and allowing users to select hardware (instruments) from a list of detected hardware (instruments) (see paragraph [0020], “*the computer system may automatically*

Art Unit: 2175

detect the one or more message-based instruments that are connected to the computer system. In other words, the computer system may automatically scan for message-based instruments coupled to the system"). It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the scanning for available hardware of Fuller with the invention of Carlson/Johnson in order to allow custom hardware components to be added to the system.

Regarding claim 8, Carlson/Johnson/Fuller teaches all the steps of claim 8 except that instructions for scanning involves instructions for receiving user-defined commands to be sent to the hardware device to attempt to identify the hardware device. Fuller teaches allowing the user to initiate a hardware scan. A user-initiated hardware scan is being interpreted with the broadest reasonable interpretation to be the same as sending user-defined command to a hardware device (see paragraph [0020], "*A user interface (UI) may be provided that allows the user to initiate a scan for message-based instruments. The user may scroll through and select an instrument from a list of detected instruments, or may otherwise specify a particular instrument to be communicated with*"). It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the user-initiated hardware scan of Fuller with the invention of Johnson in order to allow custom hardware components to be added to the system on demand.

Regarding claim 12, Carlson/Johnson/Fuller teaches every limitation of claim 12 except that at least one of instructions for providing at least one hardware object and providing at least one software object further comprises instructions for accessing at

Art Unit: 2175

least one of a hardware object and a software object located on a remote computer.

Fuller teaches that tasks associated with hardware instruments may be created and made accessible on a web site (see Fuller paragraph [0168]; “*Tasks may be collected and organized for distribution, for example through a website*”). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the invention of Fuller with the invention of Carlson/Johnson in order to allow measurement or testing over a network.

Regarding claim 13, Carlson/Johnson/Fuller teaches every limitation of claim 13 except that instructions for accessing is performed through a web page. Fuller teaches that tasks associated with hardware instruments may be created and made accessible on a web site (see Fuller paragraph [0168]; “*Tasks may be collected and organized for distribution, for example through a website*”). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the invention of Fuller with the invention of Carlson/Johnson in order to allow measurement or testing over a network.

Regarding claim 14, Carlson/Johnson/Fuller teaches every limitation of claim 14 except that instructions for accessing is performed over a network. Fuller teaches that tasks associated with hardware instruments may be created and made accessible on a web site (see Fuller paragraph [0168]; “*Tasks may be collected and organized for distribution, for example through a website*”). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the invention of

Art Unit: 2175

Fuller with the invention of Carlson/Johnson in order to allow measurement or testing over a network.

Claims 9 and 10 are rejected under 35 U.S.C. 103(a) as being unpatentable over Carlson (US 2003/0033398) *supra*, Johnson (US 2003/0001896) *supra*, and Hsiung et al. (hereinafter Hsiung), United States Patent Application Publication number 2003/0083756.

Regarding claim 9, Carlson/Johnson teaches all the elements of claim 9 except that the analysis object filters data. Hsiung teaches a system for monitoring industrial components with an analysis component that performs filtering (see Hsiung paragraph [0056]; *"The upload process takes data from the acquisition device and uploads them into the main process manager 314 for processing. Here, the data are in electronic form. In embodiments where the data has been stored in data storage, they are retrieved and then loaded into the process. Preferably, the data can be loaded onto workspace to a text file or loaded into a spread sheet for analysis. Next, the filter process 302 filters the data to remove any imperfections"*). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the filtering of data of Hsiung with the invention of Carlson/Johnson for the purpose of providing data analysis functionality.

Regarding claim 10, Carlson/Johnson teaches all the elements of claim 10 except that the analysis object plots data. Hsiung teaches a system for monitoring industrial components with an analysis component that performs plotting of data (see

paragraph [0058]; “*A baseline correction process may also find response peaks, calculate $\Delta R/R$, and plot the $\Delta R/R$ verses time stamps, where the data have been captured*”). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the plotting of data of Hsiung with the invention of Carlson/Johnson for the purpose of providing data analysis functionality.

Claim 15 is rejected under 35 U.S.C. 103(a) as being unpatentable over Carlson (US 2003/0033398) *supra*, Johnson (US 2003/0001896) *supra*, Fuller (US 2003/0035008) *supra*, and Hsiung (US 2003/0083756) *supra*.

Regarding claim 15, Carlson/Johnson/Fuller teach every limitation of claim 15 except that instructions for accessing is performed by passing commands over the network in a MATLAB environment. Hsiung teaches using MATLAB in association with the invention (see Hsiung paragraph [0534]; “*Multi-way PCA is a natural choice since PCA is already included, algorithms are available for evaluation in Matlab toolboxes, and the technique serves as a good benchmark when discussing benefits of other algorithms*”). It would have been obvious to one of ordinary skill in the art that the MATLAB environment could be used as taught by Hsiung with the invention taught by Carlson/Johnson/Fuller.

Claims 16, 17, and 27 are rejected under 35 U.S.C. 103(a) as being unpatentable over Carlson (US 2003/0033398) *supra*, Johnson (US 2003/0001896)

Art Unit: 2175

supra, and Schmit et al. (hereinafter Schmit), United States Patent Application number 2003/0004670.

Regarding claim 16, Carlson/Johnson teaches every limitation of claim 16 except instructions for modifying at least one of the hardware object and the software object. Schmit teaches a system and method for building a measurement system in which the most efficient protocol to use with each measurement device is determined and applied (see Schmit paragraph [0500]). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the hardware protocol selection system of Schmit with the invention of Carlson/Johnson for the purpose of making the measurement system more efficient.

Regarding claim 17, Carlson/Johnson/Schmit teaches that modifying specifies a protocol for use by the hardware object for communication with the hardware device (see Schmit paragraph [0500]).

Regarding claim 27, Carlson/Johnson/Schmit teaches instructions for generating an analysis object that can be used in SIMULINK (see Johnson paragraph [0101]; Johnson's invention makes use of the LabVIEW environment for generating analysis objects; see also Schmit paragraph [0619]; Schmit teaches that SIMULINK is similar in function to LabVIEW).

Claims 18-24, and 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Carlson (US 2003/0033398) *supra*, Johnson (US 2003/0001896) *supra*, Hsiung

(US 2003/0083756) *supra*, and Pike et al. (hereinafter Pike), United States Patent Application Publication number 2003/0056018.

Regarding claim 18, Carlson/Johnson/Hsiung teaches every limitation of claim 18 except that modifying modifies a value stored in an array of an array-based environment. Pike teaches a system for linking users to control instruments wherein an array-based environment can be used to change the properties of the control instruments (see Pike paragraph [0010]; “*The user may also create an object array in response to an array creation command. The object array includes as elements, a first and a second instrument object. The user may change the properties of the first and second communication channels by changing properties of the object array*”; see also Pike paragraph [0070]; “*The array-based environment 104 includes functions used by the user 30 to create an instrument object 108 through function calls 46, as well as to configure an instrument object's properties and to connect the instrument object with one of the control instruments 22*”). Pike further teaches that the graphical user interface can be used to export data to an array-based environment such as MATLAB (see paragraph [0040]; “*User 30 may send a list of requests or commands to processor 20 from the GUI 14 to establish a communication channel between the computer 12 and the control instruments 22. The user 30 does so by writing a user program 80, which resides in memory 26 of computer 12. The user program 80 may be associated with the syntax of, for example, any interpreted programming environment. An interpreted programming environment may be any proprietary program that performs mathematical computations for modeling, simulation, graphics, or data analysis related to control*”).

Art Unit: 2175

instruments, among many others. An example of an interpreted programming environment is MATLAB.RTM. from MathWorks, Inc., of Natick, Mass"). It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the array-based environment steps of Pike with the measurement systems of Carlson/Johnson/Hsiung in order to provide array-based control of the measurement devices.

Regarding claim 19, Carlson/Johnson/Hsiung/Pike teaches instructions for modifying a value stored in an array of an array-based environment, thereby modifying at least one of the hardware object and the software object (see Pike paragraphs [0010], [0070], and [0040]).

Regarding claim 20, Carlson/Johnson/Hsiung/Pike teaches instructions for exporting data from the graphical interface to an array-based environment (see Pike paragraphs [0010], [0070], and [0040]).

Regarding claim 21, Carlson/Johnson/Hsiung/Pike teaches instructions for converting user actions with the graphical interface into code (see Pike paragraphs [0010], [0070], and [0040]; Pike teaches converting user actions with the graphical interface into interpreted programming code capable of performing mathematical computations for modeling, simulation, graphics, or data analysis related to control instruments).

Regarding claim 22, Carlson/Johnson/Hsiung/Pike teaches that the code is created in a MATLAB environment (see Pike paragraphs [0010], [0070], and [0040]; Pike teaches converting user actions with the graphical interface into interpreted

programming code capable of performing mathematical computations for modeling, simulation, graphics, or data analysis related to control instruments).

Regarding claim 23, Carlson/Johnson/Hsiung/Pike teaches that the code comprises steps to create an analysis object, configure the analysis object and write and read data from the analysis object (see Pike paragraphs [0010], [0070], and [0040]; Pike teaches converting user actions with the graphical interface into interpreted programming code capable of performing mathematical computations for modeling, simulation, graphics, or data analysis related to control instruments).

Regarding claim 24, Carlson/Johnson/Hsiung/Pike teaches that the code comprises an analysis routine (see Pike paragraphs [0010], [0070], and [0040]; Pike teaches converting user actions with the graphical interface into interpreted programming code capable of performing mathematical computations for modeling, simulation, graphics, or data analysis related to control instruments).

Regarding claim 26, Carlson/Johnson/Hsiung/Pike teaches instructions for generating an analysis object so that the analysis object can be used in MATLAB (see Pike paragraph [0040]).

Claim 29 is rejected under 35 U.S.C. 103(a) as being unpatentable over Carlson (US 2003/0033398) *supra*, Johnson (US 2003/0001896) *supra*, and Phathayakorn et al. (hereinafter Phathayakorn), United States Patent number 5,986,653.

Regarding claim 29, Carlson/Johnson teaches every limitation of claim 29 except that the graphical interface comprises a tree view, wherein the tree view groups

Art Unit: 2175

the hardware objects and the software objects by a functionality characteristic. Tree views of hardware and software objects grouped by functionality were a well-known graphical user interface technique at the time the invention was made. Phathayakorn shows selecting a functional group of objects from a tree view graphical representation (see Figures 2A-5B). It would have been obvious to one of ordinary skill in the art at the time the invention was made to apply the tree view graphical representation of Phathayakorn to the invention of Carlson/Johnson in order to provide a representation of the devices on the user interface.

Claims 30-33, 36, 44-47, 50, 57, and 48 are rejected under 35 U.S.C. 103(a) as being unpatentable over Carlson (US 2003/0033398) *supra*, Johnson (US 2003/0001896) *supra*, and Gray et al. (hereinafter Gray), United States Patent 6,185,491.

Regarding claim 30, Carlson/Johnson substantially teaches a method for managing an interface, the method comprising:

providing a graphical interface (see Carlson paragraph [0087], "*FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11*") that provides interaction with an array-based environment (Gray, addressed below), a hardware device and a software device being accessible through the graphical interface, the software device being accessible to a computer, the hardware device associated with a

Art Unit: 2175

plurality of properties used to communicate with the hardware device (see Carlson paragraph [0022], “*Further provided is a method, system, and program for configuring multiple resources in the system. User selection is received of one of multiple configuration policies, wherein each configuration policy defines resources to configure and one element for each resource to configure, and wherein each element specifies configuration parameters to use to configure the resource.*”; see also Carlson paragraph [0039], “*The services may include hardware devices, software devices, application programs, storage resources, communication channels, etc.*”);

providing a first hardware object, where the first hardware object: is accessible to the computer, is depicted in the graphical interface, and interacts with the hardware device (see Carlson paragraph [0087], “*FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11. GUI panel 650 displays icons representing each resource selected from the previous panel 600 and an associated drop down menu 652, 654, 656, 658 for the selected resources.*”);

providing a first configuration of the hardware device represented by the first hardware object, the first configuration representing a collection of properties used to communicate with the hardware device and a first collection of values associated with the properties (see Carlson paragraph [0087], “*For instance, FIG. 12 illustrates the storage device drop down menu 658 showing four different possible predefined storage device element configurations, including the selected configuration. The other droop*

down menus 654 and 656 show user or default selected element configurations to include in the service configuration policy being defined. The user would use the drop down menus 652, 654, 656, and 658 to select one predefined element configuration for each selected resource to add to the service configuration policy.”);

providing a second hardware object, where the second hardware object: is accessible to the computer, is depicted in the graphical interface, and interacts with the hardware device (see Carlson paragraph [0087], “FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11. GUI panel 650 displays icons representing each resource selected from the previous panel 600 and an associated drop down menu 652, 654, 656, 658 for the selected resources.”);

providing a second configuration of the hardware device represented by the second hardware object, the second configuration representing the same collection of properties as the first configuration and a second collection of values associated with the properties, wherein at least one value of a property differs between the first configuration and the second configuration (see Carlson paragraph [0087], “For instance, FIG. 12 illustrates the storage device drop down menu 658 showing four different possible predefined storage device element configurations, including the selected configuration. The other drop down menus 654 and 656 show user or default selected element configurations to include in the service configuration policy being defined. The user would use the drop down menus 652, 654, 656, and 658 to select one

Art Unit: 2175

predefined element configuration for each selected resource to add to the service configuration policy.”);

providing at least one software object, representative of the software device, where the software object is depicted in the graphical interface, and is configured to be interactive with the software device (see Johnson paragraph [0107], “*The run-time builder may also provide various parameters to hardware and/or software resources or devices comprised in the system to configure the hardware and/or software devices in the system according to the run-time specification to allow these devices to be used during execution of the run-time 790*”);

updating the graphical interface when the first hardware object, the second hardware object, or the software object are changed in the array-based environment (see Gray, addressed below); and

displaying the hardware object and the software object to a user (see Carlson paragraph [0087], “*Each drop down menu 652, 654, 656, 658 presents available element configurations 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c the user may select to add to the service configuration policy. As discussed, the element configurations enable configuration and control over a resource according to a predefined level.*”).

Carlson/Johnson does not disclose updating the graphical interface when the hardware object or the software object are changed in the array-based environment. Gray teaches a vehicle having networked components connected in an array in which if one component is removed, the graphical interface will be updated to reflect that the component is no longer connected to the array (see Gray column 7 lines 36-46; “*When*

a device is detected as having been removed, the device interface, previously received from the device is removed from memory (1420) and the data structure is updated to remove the device entries (1430)."). It would have been obvious to one having ordinary skill in the art at the time the invention was made to allow users to physically remove components from the system and to update the graphical interface to reflect the changes as taught by Gray in the invention of Carlson/Johnson so that it would be intuitive to add or remove components to the system.

Regarding claim 31, Carlson/Johnson/Gray teaches receiving code for execution by the hardware object (see paragraph [0255]; *"Upon execution of the graphical program, the node may receive the measurement task specification as input, invoke an expert system to analyze the measurement task specification and generate a run-time specification for the measurement task in response to the analyzing, as shown in 750 and 770 of FIG. 12")*).

Regarding claim 32, Carlson/Johnson/Gray teaches that at least one additional hardware object is provided for the hardware device (see Carlson paragraph [0087]; *"FIG. 12 illustrates the storage device drop down menu 658 showing four different possible predefined storage device element configurations, including the selected configuration")*).

Regarding claim 33, Carlson/Johnson/Gray teaches that additional hardware objects are provided for a plurality of hardware devices (see Carlson paragraph [0087]; *"FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the*

resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG.

11. GUI panel 650 displays icons representing each resource selected from the previous panel 600 and an associated drop down menu 652, 654, 656, 658 for the selected resources”).

Regarding claim 36, Carlson/Johnson/Gray teaches providing an analysis object adapted to communicate with at least one of the hardware object and the software object (see Johnson paragraph [0099]; “*underlying program instructions and/or data structures which are executed by a processor (or programmable hardware element...)*”).

Regarding claim 44, Carlson/Johnson/Gray teaches a computing device comprising:

an array-based environment (see Gray column 7 lines 36-46; “*When a device is removed from its bus connection (1400), the vehicle control center detects that a device previously installed is no longer connected (1410)*”, the bus connection is the same as an array-based environment);

a storage medium for storing and a processor for processing;

a graphical interface, at least one hardware device and one software device being accessible through the graphical interface (see Carlson paragraph [0087], “*FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11*”; see also Carlson paragraph [0039], “*The services may include hardware devices,*

Art Unit: 2175

software devices, application programs, storage resources, communication channels, etc.”), the hardware device associated with a plurality of properties used to communicate with the hardware device (see Carlson paragraph [0022], “Further provided is a method, system, and program for configuring multiple resources in the system. User selection is received of one of multiple configuration policies, wherein each configuration policy defines resources to configure and one element for each resource to configure, and wherein each element specifies configuration parameters to use to configure the resource.”);

a first hardware object, where the first hardware object: is accessible to the computer, is depicted in the graphical interface, and interacts with the hardware device (see Carlson paragraph [0087], “FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11. GUI panel 650 displays icons representing each resource selected from the previous panel 600 and an associated drop down menu 652, 654, 656, 658 for the selected resources.”);

a first configuration of the hardware device represented by the first hardware object, the first configuration representing a collection of properties used to communicate with the hardware device and a first collection of values associated with the properties (see Carlson paragraph [0087], “For instance, FIG. 12 illustrates the storage device drop down menu 658 showing four different possible predefined storage device element configurations, including the selected configuration. The other droop

Art Unit: 2175

down menus 654 and 656 show user or default selected element configurations to include in the service configuration policy being defined. The user would use the drop down menus 652, 654, 656, and 658 to select one predefined element configuration for each selected resource to add to the service configuration policy.”);

a second hardware object, where the second hardware object: is accessible to the computer, is depicted in the graphical interface, and interacts with the hardware device (see Carlson paragraph [0087], “FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11. GUI panel 650 displays icons representing each resource selected from the previous panel 600 and an associated drop down menu 652, 654, 656, 658 for the selected resources.”);

a second configuration of the hardware device represented by the second hardware object, the second configuration representing the same collection of properties as the first configuration and a second collection of values associated with the properties, wherein at least one value of a property differs between the first configuration and the second configuration (see Carlson paragraph [0087], “For instance, FIG. 12 illustrates the storage device drop down menu 658 showing four different possible predefined storage device element configurations, including the selected configuration. The other drop down menus 654 and 656 show user or default selected element configurations to include in the service configuration policy being defined. The user would use the drop down menus 652, 654, 656, and 658 to select one

Art Unit: 2175

predefined element configuration for each selected resource to add to the service configuration policy.”);

a plurality of software objects, each representative of a software device accessible to the computer, where each of the software objects is depicted in the graphical interface and is configured to be interactive with the software device (see Johnson paragraph [0107], “*The run-time builder may also provide various parameters to hardware and/or software resources or devices comprised in the system to configure the hardware and/or software devices in the system according to the run-time specification to allow these devices to be used during execution of the run-time 790*”); and

a display device to display the first hardware object, the second hardware object, and the plurality of software objects to a user in a single graphical interface simultaneously (see Carlson paragraph [0087], “*FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11. GUI panel 650 displays icons representing each resource selected from the previous panel 600 and an associated drop down menu 652, 654, 656, 658 for the selected resources.*”), wherein the first hardware object, the second hardware object, and the plurality of software objects are accessible through both the array-based environment and the graphical interface (see Gray column 7 lines 36-46; “*When a device is detected as having been removed, the*

device interface, previously received from the device is removed from memory (1420) and the data structure is updated to remove the device entries (1430).").

Regarding claim 45, Carlson/Johnson/Gray teaches that the system receives code for execution by the hardware objects (see paragraph [0255]; *"Upon execution of the graphical program, the node may receive the measurement task specification as input, invoke an expert system to analyze the measurement task specification and generate a run-time specification for the measurement task in response to the analyzing, as shown in 750 and 770 of FIG. 12"*).

Regarding claim 46, Carlson/Johnson/Gray teaches that a plurality of hardware objects are provided for a single hardware device (see Carlson paragraph [0087]; *"FIG. 12 illustrates the storage device drop down menu 658 showing four different possible predefined storage device element configurations, including the selected configuration"*).

Regarding claim 47, Carlson/Johnson/Gray teaches that a plurality of hardware objects are provided for a plurality of hardware devices (see Carlson paragraph [0087]; *"FIG. 12 illustrates a GUI panel 650 in which the user may select an available element configuration 214a, b, c, 216a, b, c, 218a, b, c, 220a, b, c (FIG. 3) for each of the resources 602, 604, 606, 608, 610, 612 selected in the previous GUI panel 600 in FIG. 11. GUI panel 650 displays icons representing each resource selected from the previous panel 600 and an associated drop down menu 652, 654, 656, 658 for the selected resources"*).

Regarding claim 50, Carlson/Johnson/Gray teaches that an analysis object is provided adapted to communicate with at least one of the hardware objects and the

software objects (see Johnson paragraph [0099]; “*underlying program instructions and/or data structures which are executed by a processor (or programmable hardware element...)*”).

Regarding claim 57, Carlson/Johnson/Gray teaches that the hardware object enables communication between the graphical interface and the hardware device, and the software object enables communication between the graphical interface and the software device (see Johnson paragraph [0125]; “*a graphical user interface (GUI) may be displayed which presents information for guiding the user in specifying a measurement task. The measurement task may involve a simple measurement using a single instrument or device, or may comprise a complex measurement operation using a plurality of measurement devices. In one embodiment, at least one of the plurality of measurement devices may comprise a measurement hardware device. In another embodiment, at least one of the plurality of measurement devices may comprise a virtual measurement device*”).

Claim 58 recites a computing device having substantially the same limitations as the method of claim 57. Therefore, claim 58 is rejected under the same rationale.

Claims 34, 35, 38, 48, 49, and 52 are rejected under 35 U.S.C. 103(a) as being unpatentable over Carlson (US 2003/0033398) *supra*, Johnson (2003/0001896) *supra*, Gray (6,185,491) *supra*, and Fuller (2003/0035008) *supra*.

Regarding claim 34, Carlson/Johnson/Gray teaches every limitation of claim 34 except scanning for available hardware; and creating an additional hardware object for

Art Unit: 2175

each hardware device detected and not already associated with a hardware object.

Fuller teaches a method and apparatus for controlling an instrumentation system that automatically scans for available hardware (instruments) and allowing users to select hardware (instruments) from a list of detected hardware (instruments) (see Fuller paragraph [0020], *“the computer system may automatically detect the one or more message-based instruments that are connected to the computer system. In other words, the computer system may automatically scan for message-based instruments coupled to the system”*). It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the scanning for available hardware of Fuller with the invention of Carlson/Johnson/Gray in order to allow custom hardware components to be added to the system.

Regarding claim 35, Carlson/Johnson/Gray teaches all the steps of claim 35 except that scanning involves instructions for receiving user-defined commands to be sent to the hardware device to attempt to identify the hardware device. Fuller teaches allowing the user to initiate a hardware scan. A user-initiated hardware scan is being interpreted with the broadest reasonable interpretation to be the same as sending user-defined command to a hardware device (see Fuller paragraph [0020], *“A user interface (UI) may be provided that allows the user to initiate a scan for message-based instruments. The user may scroll through and select an instrument from a list of detected instruments, or may otherwise specify a particular instrument to be communicated with”*). It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the user-initiated hardware scan of

Art Unit: 2175

Fuller with the invention of Carlson/Johnson/Gray in order to allow custom hardware components to be added to the system on demand.

Regarding claim 38, Carlson/Johnson/Gray teaches every limitation of claim 38 except that at least one of providing at least one hardware object and providing at least one software object further comprises accessing at least one of a hardware object and a software object located on a remote computer. Fuller teaches that tasks associated with hardware instruments may be created and made accessible on a web site (see Fuller paragraph [0168]; “*Tasks may be collected and organized for distribution, for example through a website*”). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the invention of Fuller with the invention of Carlson/Johnson/Gray in order to allow measurement or testing over a network.

Claims 48, 49, and 52 recite a system with substantially the same limitations as claims 34, 35, and 38, respectively. Therefore, the claims are rejected under the same rationale.

Claims 39, 40, 43, 53, and 54 are rejected under 35 U.S.C. 103(a) as being unpatentable over Carlson (US 2003/0033398) *supra*, Johnson (2003/0001896) *supra*, Gray (US 6,185,491) *supra*, and Schmit (US 2003/0004670) *supra*.

Regarding claim 39, Carlson/Johnson/Gray teaches every limitation of claim 39 except modifying at least one of the hardware object and the software object. Schmit teaches a system and method for building a measurement system in which the most

Art Unit: 2175

efficient protocol to use with each measurement device is determined and applied (see Schmit paragraph [0500]). It would have been obvious to one of ordinary skill in the art at the time the invention was made to combine the hardware protocol selection system of Schmit with the invention of Carlson/Johnson/Gray for the purpose of making the measurement system more efficient.

Regarding claim 40, Carlson/Johnson/Gray/Schmit teaches that modifying specifies a protocol for use by the hardware object for communication with the hardware device (see Schmit paragraph [0500]).

Regarding claim 43, Carlson/Johnson/Gray/Schmidt teaches generating an analysis object that can be used in SIMULINK (see Schmit paragraph [0619]).

Claims 53 and 54 recite a system with substantially the same limitations as claims 39 and 40. Therefore, claims 53 and 54 are rejected under the same rationale.

Claims 41, 42, and 55 are rejected under 35 U.S.C. 103(a) as being unpatentable over Carlson (US 2003/0033398) *supra*, Johnson (US 2003/0001896) *supra*, Gray (US 6,185,491) *supra*, and Pike (US 2003/0056018) *supra*.

Regarding claim 41, Carlson/Johnson/Gray teaches every limitation of claim 41 except that modifying modifies a value stored in an array of an array-based environment. Pike teaches a system for linking users to control instruments wherein an array-based environment can be used to change the properties of the control instruments (see Pike paragraph [0010]; “*The user may also create an object array in response to an array creation command. The object array includes as elements, a first*

Art Unit: 2175

and a second instrument object. The user may change the properties of the first and second communication channels by changing properties of the object array”; see also Pike paragraph [0070]; “The array-based environment 104 includes functions used by the user 30 to create an instrument object 108 through function calls 46, as well as to configure an instrument object's properties and to connect the instrument object with one of the control instruments 22”). Pike further teaches that the graphical user interface can be used to export data to an array-based environment such as MATLAB (see Pike paragraph [0040]; “User 30 may send a list of requests or commands to processor 20 from the GUI 14 to establish a communication channel between the computer 12 and the control instruments 22. The user 30 does so by writing a user program 80, which resides in memory 26 of computer 12. The user program 80 may be associated with the syntax of, for example, any interpreted programming environment. An interpreted programming environment may be any proprietary program that performs mathematical computations for modeling, simulation, graphics, or data analysis related to control instruments, among many others. An example of an interpreted programming environment is MATLAB.RTM. from MathWorks, Inc., of Natick, Mass”). It would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the array-based environment steps of Pike with the measurement systems of Carlson/Johnson/Gray in order to provide array-based control of the measurement devices.

Regarding claim 42, Carlson/Johnson/Gray teaches every limitation of claim 42 except generating an analysis object so that the analysis object can be used in

Art Unit: 2175

MATLAB. Pike teaches that the graphical user interface can be used to export data to an array-based environment such as MATLAB (see Pike paragraph [0040]; “*User 30 may send a list of requests or commands to processor 20 from the GUI 14 to establish a communication channel between the computer 12 and the control instruments 22. The user 30 does so by writing a user program 80, which resides in memory 26 of computer 12. The user program 80 may be associated with the syntax of, for example, any interpreted programming environment. An interpreted programming environment may be any proprietary program that performs mathematical computations for modeling, simulation, graphics, or data analysis related to control instruments, among many others. An example of an interpreted programming environment is MATLAB.RTM. from MathWorks, Inc., of Natick, Mass*”). It would have been obvious to a person of ordinary skill in the art at the time the invention was made to provide MATLAB support as taught by Pike to the measurement systems of Carlson/Johnson/Gray in order to provide MATLAB support for the measurement devices.

Claim 55 recites a system having substantially the same limitations as claim 41. Therefore, claim 55 is rejected under the same rationale.

Response to Arguments

Applicant's arguments with respect to the claims have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

The prior art made of record and not relied upon is considered pertinent to applicant's disclosure:

- Radjabi (US 2003/0103080) System and method for developing a code generator for object-oriented communication protocol
- Rothman et al. (US 2004/0109017) Decoupled hardware configuration manager
- Asoh et al. (US 2005/0235221) Computer, display device setting method, and program

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Stephen Alvesteffer whose telephone number is (571)270-1295. The examiner can normally be reached on Monday-Friday 10:00AM-6:30PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Chat Do can be reached on (571)272-3721. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Art Unit: 2175

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Stephen Alvesteffer
Examiner
Art Unit 2171

/S. A./
Examiner, Art Unit 2171

/Chat C. Do/
Supervisory Patent Examiner, Art Unit 2171